EML - Bug #2512

require text content in elements to be non-empty

08/15/2006 10:44 AM - Matt Jones

Status:	Resolved	**Start date:**	08/15/2006
Priority:	Immediate	**Due date:**	
Assignee:	Matt Jones	**% Done:**	0%
Category:	eml - general bugs	**Estimated time:**	0.00 hour
Target version:	EML2.1.0	**Spent time:**	0.00 hour
Bugzilla-Id:	2512		

Description

Current EML schemas allow text content to be empty, which defeats validation rules by allowing users to provide content such as:
<attributeName> </attributeName>
I propose that these uses of empty strings should not be valid. We can acheive this by redefining the datatype we use for strings to have a minimum length of 1 and a pattern that requires some non-whitespace characters.

In XML Schema, we can declare the element to be of type eml:nonemptystring
where eml:nonemptystring is a simple type derived from xs:string like this:
<simpleType name='nonemptystring'>
<restriction base='string'>
<minLength value='1'/>
<pattern value='\s*(^\s)+.*'/>
</restriction>
</simpleType>

I'm not sure if that regular expression quite gets what we want, but it is close and would need some testing. It is intended to sleect (zero or more whitespace characters) followed by (one or more non-whitespace characters) followed by (any additional characters). We probably could remove the plus symbol as its redundant with the subsequent .*

History

#1 - 09/22/2008 12:20 PM - Margaret O'Brien

targeting for 2.1.0, although may drop back to unspecified.

#2 - 11/04/2008 02:09 PM - Margaret O'Brien

The pattern for this type will be something resembling:
<xs:pattern value="[\s]*[\S][\s\S]*"/>

I am assuming that we still want to allow newlines in strings, and the dot (.) specifically does not match these. At least some current xs:strings have these (e.g. <title> in test/eml-datasetWithCitation.xml).
need to test against some docs with \r\n as well

#3 - 11/08/2008 12:56 PM - Margaret O'Brien

We need to look at the effect on instance documents of switching all xs:string to NonEmptyStringType. This type-switch will probably have a bigger effect on the ability of authors to migrate their documents than the changes to the document structure itself. Structure changes will be accomplished by the xsl stylesheet, but retyping all strings means that content could now be required where none previously existed.

To start, I considered just the anonymous simple type elements that are
required by EML and are type="xs:string". It seemed reasonable that if an element was optional, that its content could also be optional. In all, there are 81 of these, which are generally easy to retype with a statement like:
sed -e '/\<xs:element\ name/{
/minOccurs=\"0\"/!s/xs:string/res:NonEmptyStringType/
}
'

There are other elements which could be examined and retyped manually, or would
be caught by a general s/xs:string/res:NonEmptyStringType/ E.g., see <keyword>
(eml-resource.xsd) -- a complexType/simpleContent, so the reference to
xs:string occurs below the element declaration. Other elements (and many
attributes) use xs:restriction base="xs:string" as the start of an enumeration
list, but changing these to base="NonEmptyStringType" seems superfluous.

So to start, only one schema file, "eml-resource.xsd", has been checked into

CVS, so that others can try out the effect of NonEmptyStringType while its scope is small. Particularly, I was thinking about Morpho. 7 element declarations occur in this file that were formerly of xs:string, and now are NonEmptyStringType. See the list below. I think that Morpho wizards deal with only title, references and keyword, although any are available in the tree editor. My local copy has all 81 (anonymous, simple) element declarations retyped (in 17 schema docs), plus the 5 anonymous attributes. I am testing a variety of EML201 documents from the LTER metacat against this schema as I convert them -- basically while I work on the XSL stylesheet.

title
distribution/connectionDefinition/parameterDefinition/name
distribution/connectionDefinition/parameterDefinition/description
distribution/connection/parameter/name
distribution/connection/parameter/description
distribution/offline/MediumName
references (multiple paths)
keyword (a named type)

#4 - 11/11/2008 04:04 PM - Jing Tao

I checked the morpho code and we use those three path at new package wizard.
title
distribution/offline/MediumName
keyword (a named type)

Morpho also checks if the the input is a empty string. If it's, morpho will ask user to input something there.

#5 - 11/21/2008 04:55 PM - Margaret O'Brien

The optional elements have had their xs:strings retyped to res:NonEmptyStringType.

#6 - 03/27/2013 02:20 PM - Redmine Admin

Original Bugzilla ID was 2512